



# Introducción al Diseño de Sistemas de Información

Unidad N° IV: Diseño de Interfaz de Usuario.



**Facultad Regional Santa Fe Universidad Tecnológica Nacional**





## DISEÑO DE INTERFACES DE USUARIO

Uno de los niveles de diseño que podemos considerar es lo que se denomina “Diseño Externo”. Este se refiere a la parte del software que está expuesta y que será, de hecho, la manera en que el usuario conocerá el sistema.

El diseño externo consiste de:

- Diseño de interfaz de usuario
- Diseño de entradas.
- Diseño de salidas.

Como contrapartida, existe lo que se denomina “Diseño Interno”, que se refiere a lo que los programadores necesitan para poder codificar los programas. Y consiste básicamente de:

- Diseño de la estructura del sistema.
- Diseño de los algoritmos de los procesos.

Vamos a ocuparnos ahora del ‘Diseño Externo’, que normalmente debe ser aprobado por el usuario, antes de realizar el diseño de los algoritmos de procesos, los cuales se verán afectados por estas definiciones.

**Diseño Externo:**  
manera en que el usuario conocerá el sistema.

## DISEÑO DE LA INTERFAZ DEL USUARIO

### Objetivos de la Interfaz de usuario

Para la mayoría de los usuarios, “la interfaz es el sistema”.

Aunque este bien o mal diseñada, permanece como una representación del sistema y por tanto es de la competencia del ingeniero de sistemas trabajar sobre esta para que la imagen que tiene el usuario esté en concordancia con la calidad interna del software.

Se debe tener como objetivo, el diseño de una interfaz que ayuden a los usuarios y a sus empresas a obtener o introducir información al sistema, y que a la vez satisfagan los siguientes objetivos:

Para el usuario, la interfaz es el sistema.

1. **Eficacia**, al lograr mediante el diseño de interfaces, que el usuario tenga acceso al sistema, de tal forma que sea congruente con sus necesidades particulares.
2. **Eficiencia** demostrada a través de interfaces que *mejoren* la velocidad de captura de los datos y reduzcan los errores.
3. **Consideración del usuario**, al demostrar un diseño adecuado de la interfaz y que favorezca la retroalimentación del sistema para los usuarios en forma apropiada.
4. **Productividad**, para que le permitan al usuario capturar un gran número de transacciones en forma rápida y efectiva.

Con esto en mente, nos adentraremos en una exposición más detallada de cada uno de los objetivos.

## TIPOS DE INTERFACES DE USUARIO

La interfaz cuenta con dos componentes principales: el *lenguaje de presentación*, que es parte de la relación computadora-hombre, y el *lenguaje de acción* que caracteriza la parte hombre-computadora. En conjunto, ambos conceptos cubren la fórmula y el contenido del término *interfaz del usuario*.

La interfaz es la comunicación o diálogo que se desarrolla entre el sistema y el usuario.

La interfaz de usuario incluye:





- La navegación dentro del sistema.
- La retroalimentación a los usuarios.
- Las consultas a las bases de datos.

De hecho, el *usuario* es el centro de atención cuando se diseña una interfaz adecuada. Y estos, pueden ser:

- **Novatos:** tiene conocimientos muy escasos en el uso de una computadora, por lo que es necesario proporcionarle mucha ayuda.
- **Intermitentes:** ha usado una computadora pero por breves e inconstantes períodos, por lo que es necesario proporcionarle ayuda en forma moderada.
- **Frecuentes:** es experto en el uso de un equipo computacional por lo que requiere muy poca ayuda, lo que el sistema debe ofrecerle es rapidez.

## Navegación dentro del sistema

Para diseñar una navegación eficiente de acuerdo al tipo de usuario, se puede elegir cualquiera de los siguientes tipos de interfaz:

- Menús
- Pregunta y respuesta
- Lenguaje Natural
- Llenado de formularios
- Lenguaje de comandos.
- Interfaz gráfica de usuarios (GUI)
- Manipulación directa.

## Menús

Una interfaz de menú permite que el usuario elija las posibles opciones de una lista en pantalla. Al responder el usuario al menú se limita a las opciones que se le presentan. El usuario no necesita conocer el sistema pero sí necesita saber que tareas pueden realizarse.

Por ejemplo, en un menú de un típico procesador de palabras, los usuarios pueden elegir entre "editar", "copiar" o "imprimir". Sin embargo, para sacar el máximo provecho del menú, los usuarios deben estar al tanto de las tareas potenciales del sistema y de lo que realmente desean hacer. En la Figura IV.1 puede verse un ejemplo de una "menú de barra" junto con un menú desplegable.

Los menús, considerados como interfaces, no son dependientes del hardware. Pero existen ciertas variantes que están relacionadas con el dispositivo de hardware utilizado para el acceso: a través del teclado, de lápices ópticos o de mouse, y eventualmente por voz.

Las opciones pueden identificarse mediante un número, una letra o una clave, pero los inexpertos en mecanografía preferirán los números a las letras, al poder ubicar con mayor rapidez las teclas de los números. En la Figura IV.2 se muestra un ejemplo de un programa orientado por menús. El usuario cuenta con siete acciones diferentes a realizar para el uso de los archivos personales. Con el fin de elegir la opción "5) Agregar un empleado", el usuario únicamente debe pulsar un 5. Esto guiará al usuario a una nueva pantalla, la cual estará esperando los datos del nuevo empleado.



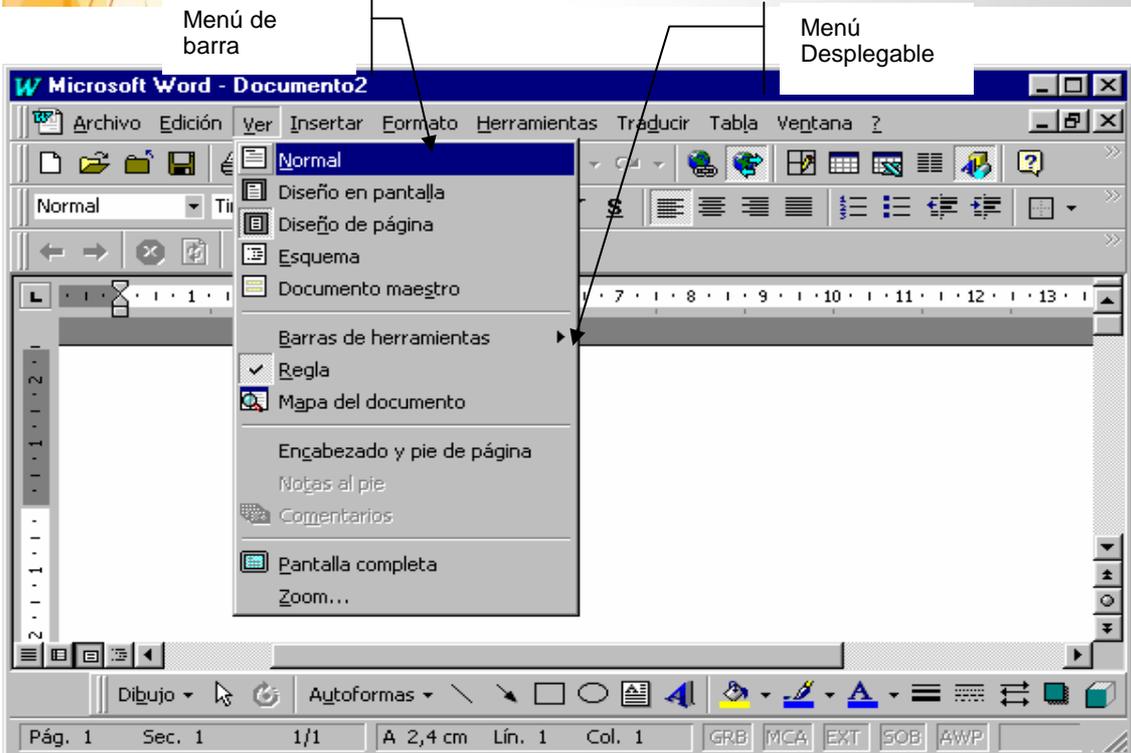


Figura IV. 1 – Ejemplo de menú de barra y menú desplegable

ARCHIVO PERSONAL

1)	Localizar el registro de un empleado por su apellido	(Ctrl-1)
2)	Localizar registro de un empleado por número de seguro social	(Ctrl-2)
3)	Imprimir empleados en orden alfabético	(Ctrl-3)
4)	Imprimir los empleados, primero por departamento y luego por orden alfabético	(Ctrl-4)
5)	Agregar un empleado	(Ctrl-5)
6)	Retirar un empleado	(Ctrl-6)
7)	Salir	(Ctrl-7)

Estilo comúnmente utilizado en interfaces caracter, y para usuarios inexpertos.

Figura IV. 2 – Un menú con ‘identificador de opciones numéricos’

También podría seleccionar una opción con las flechas y presionar <Enter>, pararse con el lápiz óptico o mouse y hacer ‘CLICK’, o saltar el menú con la tecla de acción inmediata (HOT KEY) ‘Ctrl-5’.

Al diseñar una interfaz de menú es importante mantener la *consistencia*. Con el fin de tener acceso a una sección del menú, el usuario debe presionar la tecla de RETURN o ENTER, o bien, la computadora se dirigirá a la rutina deseada cuando se presione una sola tecla. Por ejemplo, si el menú ofrece números del 0 al 9 es posible pulsar una sola vez el teclado, sin presionar el RETURN. Si se tienen más números, se necesitará de un retorno para distinguir si el usuario intenta introducir 1, 10 o 100.



Los menús pueden encontrarse *anidados*, uno dentro del otro, de tal forma que el usuario los solicite conforme los requiere (caso presentado como 'Menú Desplegable' en la Figura IV.1). Por ejemplo, el usuario comienza con el menú principal y luego selecciona la opción para 'Imprimir'. Esto lo lleva a un nuevo menú de impresión donde se muestran las opciones específicas de impresión. Si el usuario elige la opción para imprimir el archivo del cliente aparecerá una nueva pantalla con nuevas opciones detalladas para la impresión de los archivos del cliente, y así.

Los *menús anidados* tienen ciertas ventajas:

- ✓ Permiten que la pantalla se mantenga despejada, lo cual es consistente con un buen diseño de pantalla.
- ✓ Evitan que el usuario siga viendo aquellas opciones del menú que ya no son de su interés, y de tal forma, se reduce la información irrelevante para el usuario.
- ✓ Permiten un movimiento rápido a lo largo del programa.

Tener presente tanto a los usuarios inexpertos como los expertos

Aunque en general son una bendición para los inexpertos, los menús anidados pueden presentar ciertos problemas para los usuarios experimentados.

- ☞ Los usuarios experimentados pueden impacientarse al tener que elegir diferentes opciones en menús sucesivos antes de llegar a la rutina que desean.

Una forma para solucionar esto es dar como una alternativa que todas las opciones de los menús puedan invocarse mediante un solo comando de línea o una 'hot key' (tecla de acceso rápido). Esto hace que el sistema sea atractivo tanto para los usuarios expertos como para los que no lo son.

## Interfaz de preguntas y respuestas

En este tipo de interfaces la computadora plantea sobre la pantalla una pregunta al usuario. Con el fin de interactuar, el usuario por lo general proporciona una respuesta (a través de un teclado) y la computadora responderá con base en tal información de entrada de una manera pre-programada, después de ello, el cursor se desplazará a la siguiente pregunta, y así.

Numerosos desarrollos en las ciencias administrativas utilizan la interfaz de pregunta/respuesta. Como ocurre en la mayoría de las interfaces de preguntas y respuestas, el sistema de cómputo controla la secuencia de las preguntas y el usuario responde a lo que se le solicita. Por ejemplo, al responder a la frase "ahora proporcione el número de restricciones", el usuario tecléa el número correspondiente.

El ejemplo planteado en la Figura IV.3, se corresponde a una parte de un sistema de resolución de ecuaciones, donde los datos para el cálculo se ingresan uno a uno.

**Interfaz pregunta/respuesta.**

**Introduzca por favor el número de variables del problema**

> 3

**Ahora proporcione el número de restricciones**

> 4

**De las cuatro restricciones, ¿cuáles tienen variables con desviación negativa?**

> 2

**Ahora introduzca los valores del lado derecho, separados por comas**

> 100,500

**Ahora introduzca los coeficientes de los recursos para la primera ecuación, separados por comas**

Figura IV. 3 – Interfaz de preguntas - respuestas



Los programadores intentan plantear preguntas de una manera concisa y comprensible en una interfaz pregunta/respuesta, pero también deben anticiparse a los tipos de respuesta del usuario y que el sistema deberá aceptar. Una gran diversidad de respuestas se traducen de manera directa en un incremento en la complejidad de la programación.

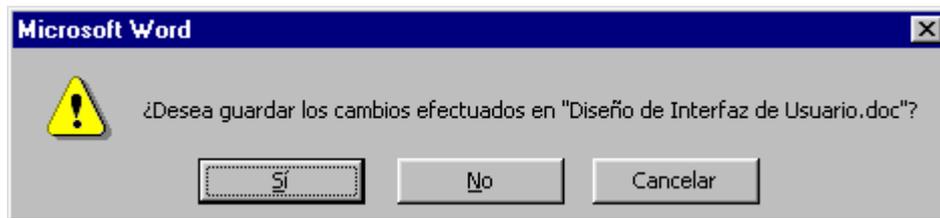
Cuando se diseñan las interfaces, se toma la decisión del *grado de flexibilidad* con que contará el usuario al momento de responder las preguntas. El usuario requiere enterarse del grado de flexibilidad que se le otorga. Por ejemplo, el usuario deberá saber si al teclear una "S" sustituye de manera aceptable a la respuesta "Si".

Conviene incluir ayuda adicional u orientación para recordarle al usuario qué respuestas serían adecuadas y sobre esta base, muchos programadores así lo hacen. Esto es importante, porque si los usuarios realizan demasiado esfuerzo adicional en la búsqueda de respuestas o en recordar la manera de contestar, podrán sentirse insatisfechos y despreciar el uso del sistema.

La experiencia indica que conforme los usuarios se involucran más en el sistema, también se llegan a impacientar con preguntas repetitivas y detalladas. Preferirían una opción que les permitiera contestar de manera breve a las preguntas.

Sin embargo, los usuarios nuevos, poco familiarizados con las computadoras considerarán bastante cómoda a la interfaz pregunta/respuesta y llegarán a crear una confianza durante su experiencia. Sin embargo, necesita estar alerta para que este nuevo usuario no sobrestime las capacidades del sistema, más de las que realmente contiene.

Un ejemplo común de interfaz de preguntas y respuestas:



Las preguntas y respuestas son comunes a todos los sistemas. Normalmente involucran 'toma de decisión'.

Figura IV. 4 – Ejemplo de interfaz de preguntas y respuestas

## Interfaz en lenguaje natural

La interfaz en lenguaje natural es quizás, el sueño y el ideal de los usuarios inexpertos, ya que permite la interacción con la computadora por medio de un lenguaje común o "natural" y no se requieren habilidades especiales del usuario cuando éste interacciona con computadoras que utilizan el lenguaje natural.

Por ejemplo, la frase que dice, "Haga una lista de todos los vendedores que hayan cubierto sus cuotas para este mes", es muy directa.

**> Haga una lista de todos los vendedores que hayan cubierto sus cuotas para este mes.**

Tom Otto  
Roz Berry  
Spin Etch

**> Compare el porcentaje de remanentes que producen cada uno de los tres almacenes.**

Fair Oaks	4%
Tyson's	5%
Metro Center	3%

**> Grafique la venta de unidades de discos, con base mensual para los últimos tres años.**

Figura IV. 5 – Ejemplo de 'lenguaje natural'



Sin embargo, si se utiliza el lenguaje natural para expresarse en el idioma típico, como se mostró en el ejemplo anterior, la esperanza de utilizarlo como medio de comunicación entre las computadoras puede llegar a ser desalentador. El idioma es ambiguo, ya que esta repleto de múltiples excepciones para cada una de las reglas.

En numerosos contextos, la ambigüedad le confiere ventajas a los comunicólogos, en particular cuando se intenta ser diplomático o no se desea dar una respuesta comprometedor. Sin embargo, los inconvenientes e irregularidades propios de las ambigüedades del idioma implican un problema de programación extremadamente complejo. Existen ciertos intentos limitados de interfaces en lenguaje natural para aplicaciones particulares donde otro tipo de interfaz sería poco conveniente. Los problemas de implicación y la demanda extraordinaria en recursos de cómputo han limitado el desarrollo de las interfaces en lenguaje natural. Sin embargo, numerosos programadores e investigadores están trabajando de manera activa en el desarrollo de interfaces en lenguaje natural, de tal forma que es un área que definitivamente continuará creciendo y por ello requerirá de un estrecho seguimiento.

Una primera aproximación a este tipo de interfaces (con las licencias del caso), son los lenguajes de programación denominados de 4GL (Lenguajes de Cuarta Generación), y los que normalmente respetan una sintaxis en lenguaje inglés.

Por ejemplo, la siguiente expresión sería un equivalente a la expresión "Haga una lista de todos los vendedores que hayan cubierto sus cuotas para este mes":

```
FOR-EACH vendedores WHERE cuota-mes >= minima-cuota:  
  DISPLAY vendedores.nombre
```

## Formularios de entrada/salida

Los formularios de entrada/salida son formularios en pantalla que despliegan campos que contienen datos o parámetros que requieren ser comunicados o requeridos al usuario. El formulario con frecuencia es un 'facsimil' de un formulario familiar para el usuario. Esta técnica de interfaz también se conoce como un método basado en formularios y de llenado de formularios. También se lo llama de "FULL SCREEN".

Los menús y las líneas de comandos pueden programarse de tal forma, que los usuarios identifiquen la aplicación deseada aún antes de aparecer el formulario. El usuario puede seleccionar formularios diferentes para propósitos diferentes.

Los formularios en pantalla muestran la información que deberá introducirse, así como su ubicación.

Los campos en blanco que requieran de la entrada de información deben resaltarse del resto de los campos (mediante caracteres en video inverso o que destellan, o algún elemento visual distintivo). En la Figura IV.6 puede verse un ejemplo de este tipo de interfaz.

El usuario deberá desplazar el cursor de campo en campo, mediante un sencillo juego de las teclas de flechas. Este juego debe permitir el movimiento hacia un campo adelante o un campo hacia atrás. Por ejemplo: ENTER o TAB pasa al campo siguiente; SHIFT-TAB regresa al campo anterior; Flechas de cursor (arriba –abajo) pasan al campo anterior o siguiente.

Los formularios por pantalla para la entrada de datos pueden simplificarse al proporcionar valores por omisión (default) en cada uno de los campos y permitir que el usuario modifique tales valores si lo considera necesario. Por ejemplo, un sistema de administración de bases de datos diseñado para mostrar un formulario para la captura de cheques, proporciona el siguiente número de secuencia del cheque como valor por omisión cuando presenta una nueva forma en pantalla. Si el número de los cheques no fuera consecutivo, el usuario podría modificar tal valor por omisión y registrar el valor real.

La captura de campos en pantalla puede restringir los elementos alfanuméricos, de tal forma que los usuarios sólo proporcionen cifras en un campo que requiere del número del documento o capturen sólo letras en donde se solicita el nombre de la persona. Si se diera el caso de que capturaran números en donde sólo se aceptan letras, el sistema le indicará al usuario que el campo se esta llenando de manera incorrecta.



La ventaja principal de un formulario de entrada/salida es que la versión impresa del formulario, una vez llenada, sirve como un excelente elemento de documentación. Muestra tanto el nombre de los campos, así como el contenido de la captura.

Los formularios de entrada/salida cuentan con ciertas desventajas.

Este es uno de los tipos de interfaces más comúnmente utilizados.





El principal inconveniente es semejante al del problema del usuario experimentado mencionado en el caso para las interfaces de pregunta/respuesta y de menú. Los usuarios experimentados pueden llegar a impacientarse con los formularios de entrada/salida y querrán contar con una manera más eficiente para capturar los datos.

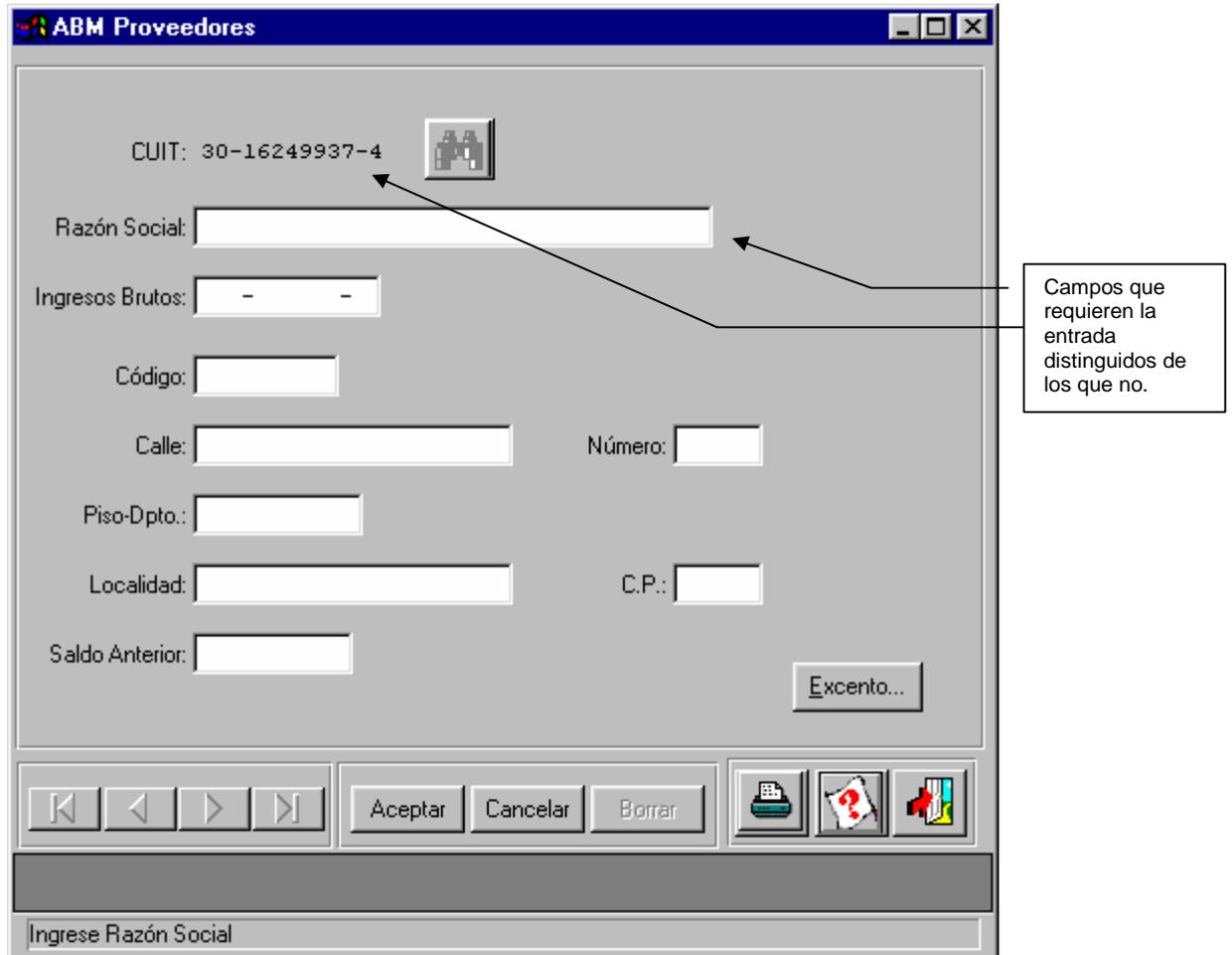


Figura IV. 6 – Ejemplo de interfaz de formulario

### Interfaz en lenguaje de comandos

La interfaz del lenguaje de comandos permite que el usuario tome el control de la aplicación mediante una serie de teclas, comandos, frases o cierta secuencia de los mismos. Es un interfaz muy popular ya que es más depurada que las que expusimos con anterioridad. Por ejemplo el DOS tiene una interfaz de lenguaje de comandos (dir, del, format, etc), al igual que Linux y Unix, los que eventualmente son 'enmascarados' por una Interfaz Gráfica de Usuario (Windows en el caso de DOS, Genome, KDE, etc., en el caso de Linux).

Los Sistemas Operativos utilizan interfaz de comandos.

Los lenguajes por comandos manipulan a la computadora como un instrumento que permite que el usuario controle el diálogo. Por lo tanto, el lenguaje por comandos confiere mayor flexibilidad y control al usuario.

Los lenguajes por comandos requieren de la memorización de reglas de sintaxis, que puede ser un obstáculo para usuarios inexpertos (sin embargo, con la práctica cualquiera puede aprenderlos). Por ejemplo, si en DOS se quiere obtener una lista ordenada por fecha de modificación, deberá utilizar el comando "DIR /O-D".

Los usuarios experimentados prefieren los lenguajes de comandos.

Otras interfaces se parecen más a las interlocuciones humanas, y en consecuencia, son las más fáciles de asimilar, incluso sin percatarse de la manera de cómo funciona una



computadora. Los usuarios experimentados tienden a preferir el lenguaje por comandos, quizás por ser mucho más rápido de operar.

## Interfaz Gráfica de Usuario (GUI)

La interfaz gráfica de usuario está relacionada fundamentalmente con la tecnología utilizada y el soporte que brinde el sistema operativo o software de base utilizado (por ejemplo: Windows o KDE para Linux).

La principal diferencia que introduce este tipo de interfaces es el uso de *iconos*. Los iconos son símbolos o figuras que representan las acciones que el sistema puede realizar. Por ejemplo: la figura de un impresora representa la acción de impresión del sistema, la figura de un disquete significa guardar un archivo en disco, etc..

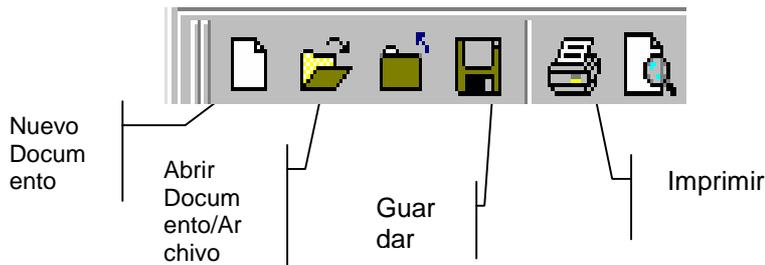


Figura IV. 7 – Iconos gráficos estándares

👍 Los iconos definen estándares que son utilizadas por todas las aplicaciones, y que son reconocidos fácilmente por los usuarios.

## Interfaz de manejo directo

Este tipo de interfaz se deriva de la GUI. El *manejo directo* le confiere al usuario en forma *gráfica*, una metáfora sólida de la aplicación. Permite el manejo directo de la representación gráfica en la pantalla como lo haría comúnmente con el objeto físico, el cual puede realizarse mediante el acceso por el teclado, ratón o joystick.

El manejo directo requiere de una mayor sofisticación del sistema, con respecto a las interfaces estudiadas con anterioridad. Esta es la base para la *realidad virtual*.

Algunos ejemplos de interfaces de manipulación directa son: las calculadoras, la papelera de reciclaje, los equalizadores musicales, juegos de video, etc.

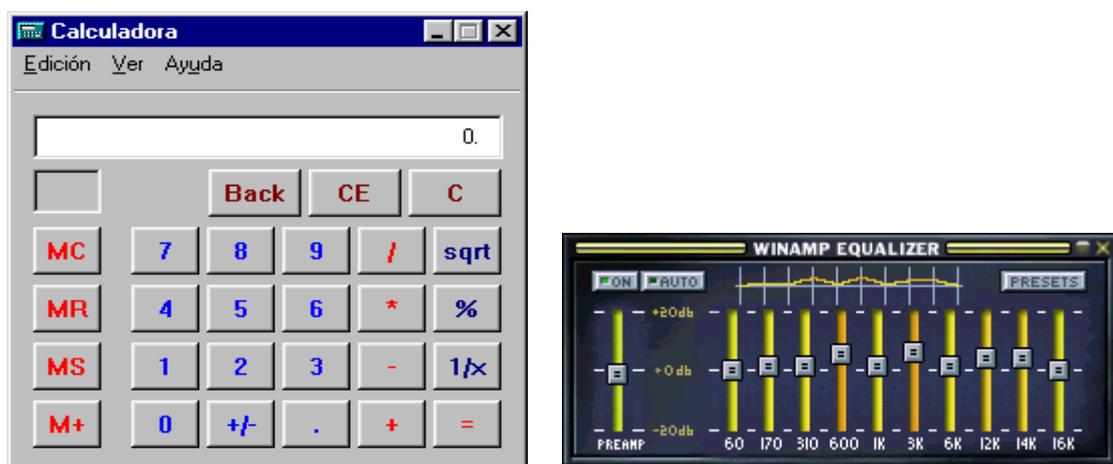


Figura IV. 8 – Ejemplos de interfaces de manipulación directa



## Otras interfaces del usuario

Existen otras interfaces del usuario que, aunque son menos comunes que las descritas anteriormente, su popularidad no deja de crecer. Dentro de estas interfaces tenemos los dispositivos de señalamiento, tales como los lápices ópticos, las pantallas sensibles al tacto y los sistemas de reconocimiento y síntesis del lenguaje. Cada una de estas interfaces cuenta con atributos especiales que las hacen únicas para aplicaciones particulares.

El dispositivo de señalamiento (con frecuencia el lápiz óptico) se utiliza para señalar sobre la pantalla un elemento de la misma. Los lápices ópticos son más comunes en Japón, donde las entradas por el teclado de miles de caracteres complejos sólo para el nombre de las personas son tareas de características terribles. Ciertas aplicaciones de ingeniería permiten el uso de un lápiz óptico para dibujar sobre la pantalla o seleccionar un elemento el menú.

Las pantallas sensibles al tacto permiten que el usuario active la pantalla mediante un ícono (u objeto) y no requieren de una experiencia especial del usuario, están autocontenidas, sin requerir de ningún dispositivo adicional para el acceso.

## El reconocimiento de la voz

El reconocimiento de la voz ha sido el sueño de los científicos y asimismo, de los escritores de ciencia ficción. Es muy atractivo pues intenta semejarse a la comunicación humana.

Con el sistema de reconocimiento de la voz, el usuario le habla a la computadora y ésta es capaz de reconocer las señales orales individuales, convirtiéndolas y aceptándolas como entradas. Hoy en día, se encuentran en operación muchos sistemas basados en el reconocimiento de la voz.

Una ventaja de los sistemas de reconocimiento de la voz es que mediante su uso, se reduce de forma considerable la duración de la captura de datos, además de que libera las manos del usuario para otras tareas. Debido a la amplia fluctuación del patrón de la voz humana, los sistemas de reconocimiento general de la voz, son más difíciles de construir, que aquellos que distinguen el tono de voz de un individuo en particular.

La siguiente tabla muestra la relación entre los tipos de usuarios e interfaces:

Tipo de Interfaz	Interfaz recomendable para:		
	Usuario Novato	Usuario Intermitente	Usuario Frecuente
Menús	✓	✓	
Pregunta y respuesta	✓	✓	
Lenguaje Natural	✓	✓	
Llenado de Formas	✓	✓	✓
Lenguaje de Comandos			✓
Interfaz Gráfica	✓	✓	✓
Manipulación Directa	✓	✓	✓

## Evaluación de las interfaces

Al evaluar las interfaces, dentro de las cuales tomará una decisión deberá tener en mente ciertas normas:

1. La *capacitación* indispensable de los usuarios debe ser aceptablemente breve.
2. Los usuarios, deben ser capaces de hacer uso de los comandos sin tener que pensar explícitamente en ellos, o sin tener que referirse a un menú de ayuda o a un manual.
3. La interfaz debe ser "*consistente*", de tal forma que se reduzcan los errores y aquellos que se presenten no se deban a un diseño deficiente.



4. El tiempo necesario para las respuestas debe ser breve para que los usuarios del sistema superen los errores.
5. Los usuarios esporádicos deben tener capacidad para comprender con rapidez el sistema.
6. Su costo no debe desvirtuar la naturaleza del sistema.

Existen interfaces diferentes, y por ello es importante percatarse de que una interfaz efectiva redundará en un compromiso duradero del usuario. Los usuarios deben desear el uso del sistema. Veremos ahora la importancia de ofrecer una retroalimentación al usuario, con el fin de soportar y sustentar su relación con el sistema.

### Retroalimentación para el usuario

Todos los sistemas requieren de la *retroalimentación* con el fin de supervisar y modificar la conducta del usuario. La retroalimentación compara la conducta presente con las metas preestablecidas y devuelve información que describe el distanciamiento existente entre el desempeño real y el que se intenta.

Ya que los seres humanos considerados como sistemas son complejos, requieren de la retroalimentación de otros para satisfacer sus necesidades psicológicas. La retroalimentación también incrementa la confianza humana. El grado de retroalimentación que se requiere es una característica personal.

Cuando los usuarios interactúan con las máquinas también necesitan de la retroalimentación para saber la manera en que progresa su trabajo. Los ingenieros de sistemas, como diseñadores de interfaces para los usuarios, necesitan percatarse de la necesidad humana de retroalimentación e incorporarla dentro del sistema. Cuando tratamos el tema de la retroalimentación para el usuario, nos estamos refiriendo de manera típica a la *retroalimentación que ofrecen los sistemas en línea*.

Aquella retroalimentación que es escasa o inoportuna, carece de utilidad, ya que sólo podemos procesar una cantidad limitada de información.

La retroalimentación para el usuario del sistema es necesaria en siete situaciones particulares, las que detallamos a continuación.

La retroalimentación es una necesidad humana.

La retroalimentación sirve para que el usuario, se entere de que:

1. La computadora aceptó la entrada
2. La entrada se encuentra en la forma correcta
3. La entrada no se encuentra en la forma correcta
4. Habrá un retraso en el procesamiento
5. La solicitud ha sido concluida
6. La computadora es incapaz de concluir la petición
7. Se dispone de mayor detalle en la retroalimentación (e indica la manera de obtenerla)

### 1. Percatarse de la aceptación de la entrada

La primera situación en la cual los usuarios requieren de la retroalimentación, es *cuando necesitan saber que la computadora aceptó la entrada*.

Por ejemplo, cuando un usuario captura un nombre, el cursor de la computadora avanza un carácter cada vez que se incorporan de manera correcta las letras, retroalimentando de esta manera al usuario.

### 2. Reconocer que la entrada se apeg a la forma correcta

El usuario necesita de la retroalimentación para entender que la entrada cuenta con la forma correcta, como ejemplo tenemos a un usuario que ejecuta un comando y la computadora lo retroalimenta con "READY", conforme avanza al siguiente punto.

Un ejemplo pobre de retroalimentación sería indicar mediante un "INPUT OK", que la entrada fue correcta, ya que esto requiere de espacio adicional, se encuentra abreviado y no le indica qué acción deberá realizar a continuación.

Los 'beeps' pueden inducir malhumor en el usuario.





### 3. Avisar que la entrada no se encuentra en la forma correcta

La retroalimentación es necesaria para advertir al usuario que la entrada carece de la forma correcta. Cuando los datos son incorrectos, una forma de informarle al usuario es presentar una ventana que describe con brevedad el problema de la entrada y la manera en que el usuario puede corregirlo.

Recuerde que el mensaje referente al error sobre la entrada debe ser *diplomático y conciso*, pero no abreviado; de tal forma que los usuarios sin experiencia son capaces de comprenderlo. La entrada es incorrecta, pero la retroalimentación no recalca el hecho de que el usuario cometió un error. Más bien, *ofrece alternativas de tal forma que el error se corrija con facilidad*.

Hasta aquí hemos expuesto facetas *visuales* de retroalimentación, pero muchos otros sistemas cuentan con una retroalimentación *auditiva*. Cuando un usuario introduce datos de manera incorrecta, tal y como se mostró con anterioridad para el error sobre el campo, el sistema puede activar una alarma, en lugar de presentar una ventana. Las alarmas, las quejas o "beeps" pueden inducir cierta frustración o malhumor en el usuario. Además, la retroalimentación sonora, por sí sola no es descriptiva, de tal forma que no ofrece ningún auxilio al usuario sobre que acciones tomar.



**Utilice de manera ocasional la retroalimentación sonora, quizás para denotar situaciones urgentes.**

### 4. Explicar un retraso en el proceso

Uno de los tipos más importantes de retroalimentación consiste en informar al usuario que habrá un retraso asociado al procesamiento de los datos tal que el usuario no solicite comandos que impidan o bloqueen el proceso.

Un mensaje típico es:

"Procesando datos ....."

o

"Procesando datos. Espere por favor ....."

o

las representaciones gráficas del tipo 'barras de avance' o el 'reloj de arena' de Windows.

Avisar que habrá retraso es muy importante.

### 5. Aviso de que su petición fue llevada a cabo por completo

Los usuarios necesitan saber cuándo fue llevada a cabo por completo su petición para poder hacer nuevas peticiones. Si en una pantalla el usuario ha solicitado la preparación de un índice ordenado por el nombre, cuando concluye la indexación, el sistema debe contestar con "EJECUTADO". El usuario ahora sabe que puede proceder con seguridad.

Esto está en relación con el punto anterior, donde se avisa que espere, pero una vez finalizado el proceso, se notifica de este hecho.

### 6. Aviso de que una petición no fue llevada a cabo por completo

La retroalimentación también es necesaria para indicar al usuario que la computadora es incapaz de llevar a cabo por completo una petición. El usuario puede regresar y verificar si la petición se planteó de manera correcta en lugar de continuar incorporando comandos que no pueden ejecutarse.

### 7. Retroalimentación con mayor detalle

En relación con esto, los usuarios necesitan asegurarse de que se dispone de una mayor retroalimentación, y enterarse de la manera de obtenerla. Estos casos normalmente se ven reflejados en mensajes del tipo 'Más información...' o 'Ayuda', y se corresponden, generalmente, a los casos donde se han presentado '*problemas*' (en las situaciones exitosas normalmente no son necesarias).

La retroalimentación debe diseñarse con mucho cuidado, y no debe ser exagerada, ya que sino provocaría una reacción negativa en el usuario.





***El éxito del sistema que diseñe depende de la aceptación y del grado en que se involucre al usuario. Además, es de gran importancia considerar de manera sistemática a los usuarios y no como un elemento adicional del análisis de sistemas.***

Bibliografía:

- (a) "Análisis y Diseño de Sistemas" – Kenneth E. Kendall y Julie E. Kendall – Ed. Prentice-Hall Hispanoamericana (1991)